AMENDMENT TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application.

Listing of Claims:

1-18. (canceled)

19. (currently amended) A method to test a cache comprising:

decoding a transaction address to access the cache as a memory transaction, but in which a state of a predetermined bit or bits of the transaction address is decoded as within a direct access address space to differentiate a direct access transaction from ~~a~~ the memory transaction, in order to directly access the cache;

asserting a direct access signal in response to the decoding of the ~~direct access transaction~~ predetermined bit or bits;

performing a direct access transaction of ~~a~~ the cache when the direct access signal is asserted to directly select a cache entry for the transaction, the direct access transaction overriding a hit or miss protocol used with the cache when memory transactions are to be performed, wherein the direct access transaction uses a bit field read as a tag in the memory transaction as a control field when accessing the cache in the direct access transaction; and

performing a test by using the selected entry accessed by the direct access transaction.

20. (currently amended) The method as recited in claim 19 wherein the performing the direct access transaction includes ~~selecting~~ using the control field to select a particular way for a next eviction and the performing the test includes performing a selected memory transaction which results in a cache miss evicting an entry from the particular way.

21. (previously presented) The method as recited in claim 20 wherein the performing the test includes performing a read memory transaction, in which data from memory is

cached into the entry of the particular way.

22. (previously presented) The method as recited in claim 21 further comprising performing a second direct access transaction to access the entry of the particular way and comparing the cached data to the data in memory.

23. (previously presented) The method as recited in claim 20 wherein the performing the test includes performing a write memory transaction, in which data to be stored in memory is cached into the entry of the selected way.

24. (previously presented) The method of claim 23 further comprising performing a second direct access transaction to access the entry of the particular way and comparing the cached data to the data originally selected to be written.

25. (previously presented) The method of claim 20 wherein the performing the test includes performing a memory transaction of test data that results in a miss in the cache and in which the miss causes an eviction and caches the test data into the entry of the particular way.

26. (previously presented) The method of claim 25 further comprising performing a second direct access transaction to access the entry of the particular way and comparing the cached data to the test data for error.

27. (original) The method of claim 26 further including reading a tag of the selected entry with a tag stored in a tag register to compare the two tags for error.

28. (currently amended) A method to reset a cache comprising:

decoding a transaction address to access the cache as a memory transaction, but in which a state of a predetermined bit or bits of the transaction address is decoded as within a direct access address space to differentiate a direct access transaction from ~~a~~ the memory transaction, in order to directly access the cache;

asserting a direct access signal in response to the decoding of the ~~direct access transaction~~ predetermined bit or bits;

performing a direct access transaction of ~~a~~ the cache when the direct access signal is asserted to directly select a cache entry for the transaction, the direct access transaction overriding a hit or miss protocol used with the cache when memory transactions are to be performed, wherein the direct access transaction uses a bit field read as a tag in the memory transaction as a control field when accessing the cache in the direct access transaction and the direct access transaction setting an indication that the selected entry is invalid; and

performing a memory transaction to generate a cache miss and to have a predetermined data written into the selected entry by eviction of invalid data to store predetermined data as reset data for the selected entry.

29. (currently amended) The method of claim 28 wherein the performing the direct access transaction includes ~~selecting a selected index and~~ using the control field to select a way of a cache line for eviction.

30. (original) The method of claim 29 wherein the performing the direct access and the performing the memory transaction are repeated for entries of the cache to store respective predetermined data in the cache to reset the cache to a known state.

31. (currently amended) A method to synchronize a cache comprising:

decoding a transaction address to access the cache as a memory transaction, but in which a state of a predetermined bit or bits of the transaction address is decoded as within a direct access address space to differentiate a direct access transaction from ~~a~~ the memory transaction, in order to directly access the cache;

asserting a direct access signal in response to the decoding of the ~~direct access transaction~~ predetermined bit or bits;

performing a direct access transaction of ~~a~~ the cache when the direct access signal is asserted to directly select a cache way to initialize a replacement procedure that is used for cache misses, the direct access transaction used to override a hit or miss protocol used

with the cache when memory transactions are to be performed, wherein the direct access transaction uses a bit field read as a tag in the memory transaction as a control field when accessing the cache in the direct access transaction and the control field selecting the cache way; and

performing subsequent memory transactions, in which way replacement is synchronized to commence from a known initialized way.

32. (original) The method of claim 31 further comprising writing test data into cache starting from the known initialized way.

33. (currently amended) A method to flush a cache comprising:

decoding a transaction address to access the cache as a memory transaction, but in which a state of a predetermined bit or bits of the transaction address is decoded as within a direct access address space to differentiate a direct access transaction from ~~a~~ the memory transaction, in order to directly access the cache;

asserting a direct access signal in response to the decoding of the ~~direct access transaction~~ predetermined bit or bits;

performing a direct access transaction of ~~a~~ the cache when the direct access signal is asserted to directly select a cache entry to set the cache entry for eviction when cache misses, the direct access transaction used to override a hit or miss protocol used with the cache when memory transactions are to be performed, wherein the direct access transaction uses a bit field read as a tag in the memory transaction as a control field when accessing the cache in the direct access transaction; and

performing a memory transaction to the cache entry, in which the entry is flushed due to the eviction.

34. (currently amended) The method of claim 33 wherein the performing the direct access transaction ~~selects an index and~~ uses the control field to select a way of the cache and the performing the memory transaction flushes a cache line identified by the memory transaction to memory.

35. (original) The method of claim 34 wherein the cache entry is set for eviction by setting the entry as invalid.

36. (original) The method of claim 34 wherein the cache entry is set for flushing by setting the entry as dirty.